```python
# Import Data Exploratory Packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

        #Initial Data Explorations
# Read DataFrame
df = pd.read_csv("SampleSuperstore.csv")
df
```

```
           Ship Mode    Segment        Country             City
State  \
0       Second Class   Consumer  United States        Henderson
Kentucky
1       Second Class   Consumer  United States        Henderson
Kentucky
2       Second Class  Corporate  United States      Los Angeles
California
3     Standard Class   Consumer  United States  Fort Lauderdale
Florida
4     Standard Class   Consumer  United States  Fort Lauderdale
Florida
...              ...        ...            ...              ...
...
9989    Second Class   Consumer  United States            Miami
Florida
9990  Standard Class   Consumer  United States       Costa Mesa
California
9991  Standard Class   Consumer  United States       Costa Mesa
California
9992  Standard Class   Consumer  United States       Costa Mesa
California
9993    Second Class   Consumer  United States      Westminster
California

      Postal Code Region         Category Sub-Category     Sales
Quantity  \
0           42420  South        Furniture    Bookcases  261.9600
2
1           42420  South        Furniture       Chairs  731.9400
3
2           90036   West  Office Supplies       Labels   14.6200
2
3           33311  South        Furniture       Tables  957.5775
5
4           33311  South  Office Supplies      Storage   22.3680
2
...           ...    ...              ...          ...       ...
...
```

```
9989        33180  South        Furniture  Furnishings   25.2480
3
9990        92627   West        Furniture  Furnishings   91.9600
2
9991        92627   West       Technology       Phones  258.5760
2
9992        92627   West  Office Supplies        Paper   29.6000
4
9993        92683   West  Office Supplies   Appliances  243.1600
2

      Discount    Profit
0         0.00   41.9136
1         0.00  219.5820
2         0.00    6.8714
3         0.45 -383.0310
4         0.20    2.5164
...        ...       ...
9989      0.20    4.1028
9990      0.00   15.6332
9991      0.20   19.3932
9992      0.00   13.3200
9993      0.00   72.9480

[9994 rows x 13 columns]

df.head()

        Ship Mode    Segment        Country             City
State  \
0    Second Class   Consumer  United States        Henderson
Kentucky
1    Second Class   Consumer  United States        Henderson
Kentucky
2    Second Class  Corporate  United States      Los Angeles
California
3  Standard Class   Consumer  United States  Fort Lauderdale
Florida
4  Standard Class   Consumer  United States  Fort Lauderdale
Florida

   Postal Code Region         Category Sub-Category     Sales
Quantity  \
0        42420  South        Furniture    Bookcases  261.9600
2
1        42420  South        Furniture       Chairs  731.9400
3
2        90036   West  Office Supplies       Labels   14.6200
2
3        33311  South        Furniture       Tables  957.5775
```

```
5
4        33311  South  Office Supplies      Storage   22.3680
2

   Discount    Profit
0      0.00   41.9136
1      0.00  219.5820
2      0.00    6.8714
3      0.45 -383.0310
4      0.20    2.5164

df.tail()

           Ship Mode   Segment        Country         City       State
\
9989    Second Class  Consumer  United States        Miami     Florida

9990  Standard Class  Consumer  United States   Costa Mesa  California

9991  Standard Class  Consumer  United States   Costa Mesa  California

9992  Standard Class  Consumer  United States   Costa Mesa  California

9993    Second Class  Consumer  United States  Westminster  California


      Postal Code Region         Category Sub-Category    Sales
Quantity  \
9989        33180  South        Furniture  Furnishings   25.248
3
9990        92627   West        Furniture  Furnishings   91.960
2
9991        92627   West       Technology       Phones  258.576
2
9992        92627   West  Office Supplies        Paper   29.600
4
9993        92683   West  Office Supplies   Appliances  243.160
2

      Discount   Profit
9989       0.2   4.1028
9990       0.0  15.6332
9991       0.2  19.3932
9992       0.0  13.3200
9993       0.0  72.9480

# DataFrame Information
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
```

```
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Ship Mode     9994 non-null   object 
 1   Segment       9994 non-null   object 
 2   Country       9994 non-null   object 
 3   City          9994 non-null   object 
 4   State         9994 non-null   object 
 5   Postal Code   9994 non-null   int64  
 6   Region        9994 non-null   object 
 7   Category      9994 non-null   object 
 8   Sub-Category  9994 non-null   object 
 9   Sales         9994 non-null   float64
 10  Quantity      9994 non-null   int64  
 11  Discount      9994 non-null   float64
 12  Profit        9994 non-null   float64
dtypes: float64(3), int64(2), object(8)
memory usage: 1015.1+ KB

# Check Data Shape
df.shape

(9994, 13)

# Empty Cell Check
df.isnull().sum()

Ship Mode       0
Segment         0
Country         0
City            0
State           0
Postal Code     0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64

#Column Consistence Check
df.columns

Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Postal
Code',
       'Region', 'Category', 'Sub-Category', 'Sales', 'Quantity',
'Discount',
       'Profit'],
      dtype='object')
```

```
#Remove Duplicate
New_df = df.drop_duplicates()
New_df

           Ship Mode    Segment        Country             City  
State  \
0       Second Class   Consumer  United States        Henderson  
Kentucky   
1       Second Class   Consumer  United States        Henderson  
Kentucky   
2       Second Class  Corporate  United States      Los Angeles  
California   
3     Standard Class   Consumer  United States  Fort Lauderdale  
Florida   
4     Standard Class   Consumer  United States  Fort Lauderdale  
Florida   
...              ...        ...            ...              ...  
...   
9989    Second Class   Consumer  United States            Miami  
Florida   
9990  Standard Class   Consumer  United States       Costa Mesa  
California   
9991  Standard Class   Consumer  United States       Costa Mesa  
California   
9992  Standard Class   Consumer  United States       Costa Mesa  
California   
9993    Second Class   Consumer  United States      Westminster  
California   

      Postal Code Region         Category Sub-Category      Sales  
Quantity  \
0           42420  South        Furniture    Bookcases   261.9600  
2   
1           42420  South        Furniture       Chairs   731.9400  
3   
2           90036   West  Office Supplies       Labels    14.6200  
2   
3           33311  South        Furniture       Tables   957.5775  
5   
4           33311  South  Office Supplies      Storage    22.3680  
2   
...           ...    ...              ...          ...        ...  
...   
9989        33180  South        Furniture  Furnishings    25.2480  
3   
9990        92627   West        Furniture  Furnishings    91.9600  
2   
9991        92627   West       Technology       Phones   258.5760  
2   
9992        92627   West  Office Supplies        Paper    29.6000  
```

```
4
9993        92683   West  Office Supplies   Appliances  243.1600
2

      Discount    Profit
0         0.00   41.9136
1         0.00  219.5820
2         0.00    6.8714
3         0.45 -383.0310
4         0.20    2.5164
...        ...       ...
9989      0.20    4.1028
9990      0.00   15.6332
9991      0.20   19.3932
9992      0.00   13.3200
9993      0.00   72.9480

[9977 rows x 13 columns]
```

```
# New DataFrame Shape
New_df.shape
```

```
(9977, 13)
```

```
# Unique Values in Categorical Columns
unique_categories = df['Category'].nunique()
unique_categories
```

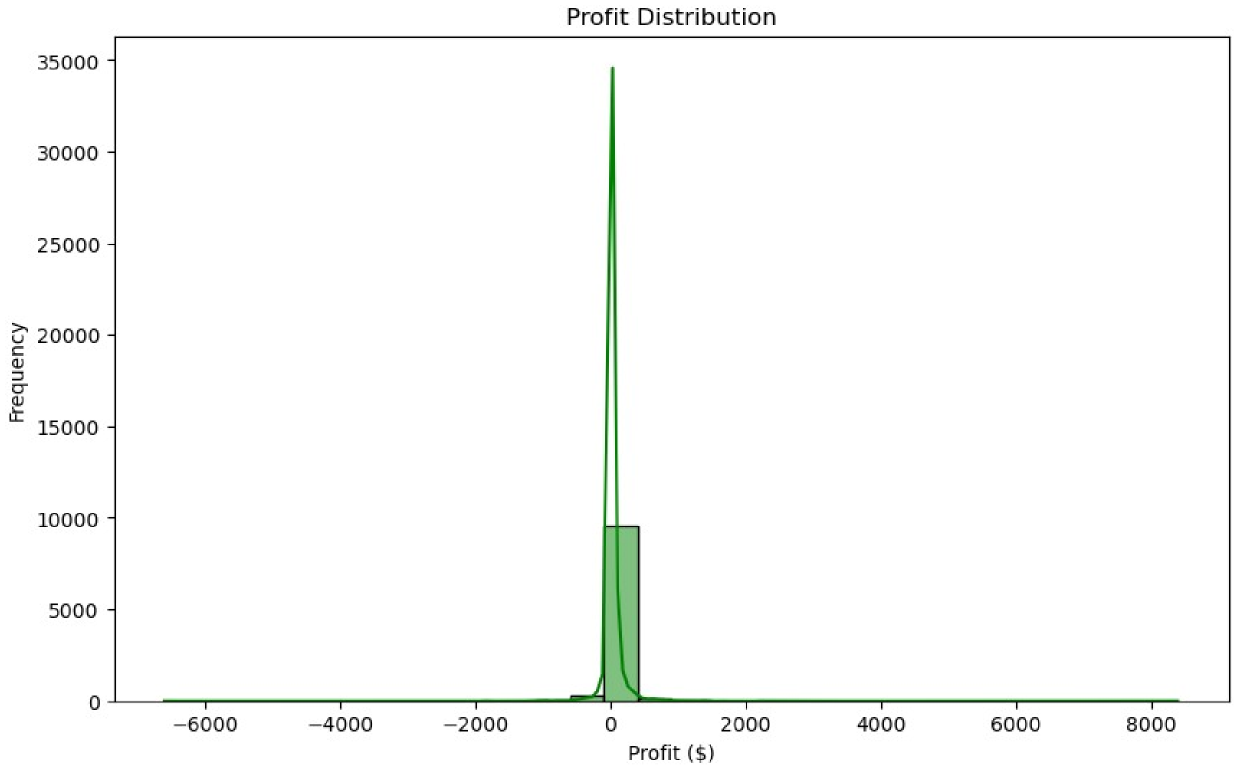
```
3
```

```
#Data Types Check
New_df.dtypes
```

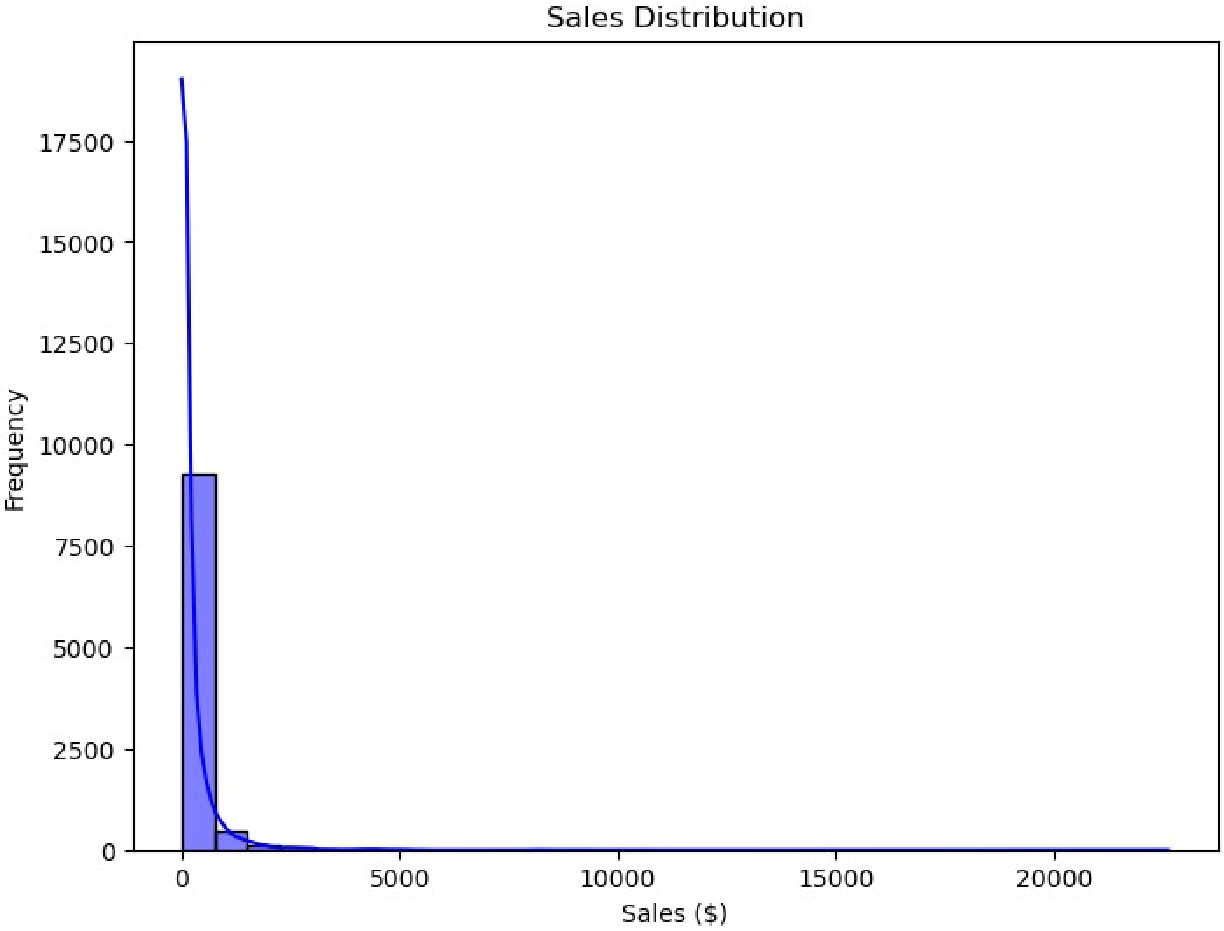
```
Ship Mode        object
Segment          object
Country          object
City             object
State            object
Postal Code       int64
Region           object
Category         object
Sub-Category     object
Sales           float64
Quantity          int64
Discount        float64
Profit          float64
dtype: object
```

```
# Data Descriptions
New_df.describe()
```

```
        Postal Code         Sales     Quantity     Discount
Profit
count   9977.000000   9977.000000  9977.000000  9977.000000
9977.00000
mean   55154.964117    230.148902     3.790719     0.156278
28.69013
std    32058.266816    623.721409     2.226657     0.206455
234.45784
min     1040.000000      0.444000     1.000000     0.000000 -
6599.97800
25%    23223.000000     17.300000     2.000000     0.000000
1.72620
50%    55901.000000     54.816000     3.000000     0.200000
8.67100
75%    90008.000000    209.970000     5.000000     0.200000
29.37200
max    99301.000000  22638.480000    14.000000     0.800000
8399.97600

#Total Sales and Profit
total_sales = New_df['Sales'].sum()
total_profit = New_df['Profit'].sum()
print(f"Total Sales: ${total_sales:.2f}, Total Profit: $
{total_profit:.2f}")

Total Sales: $2296195.59, Total Profit: $286241.42

# Histogram of Profit
plt.figure(figsize=(10, 6))
sns.histplot(df["Profit"], bins=30, kde=True, color="green")
plt.title("Profit Distribution")
plt.xlabel("Profit ($)")
plt.ylabel("Frequency")
plt.savefig("Profit Distribution")
plt.show()
```



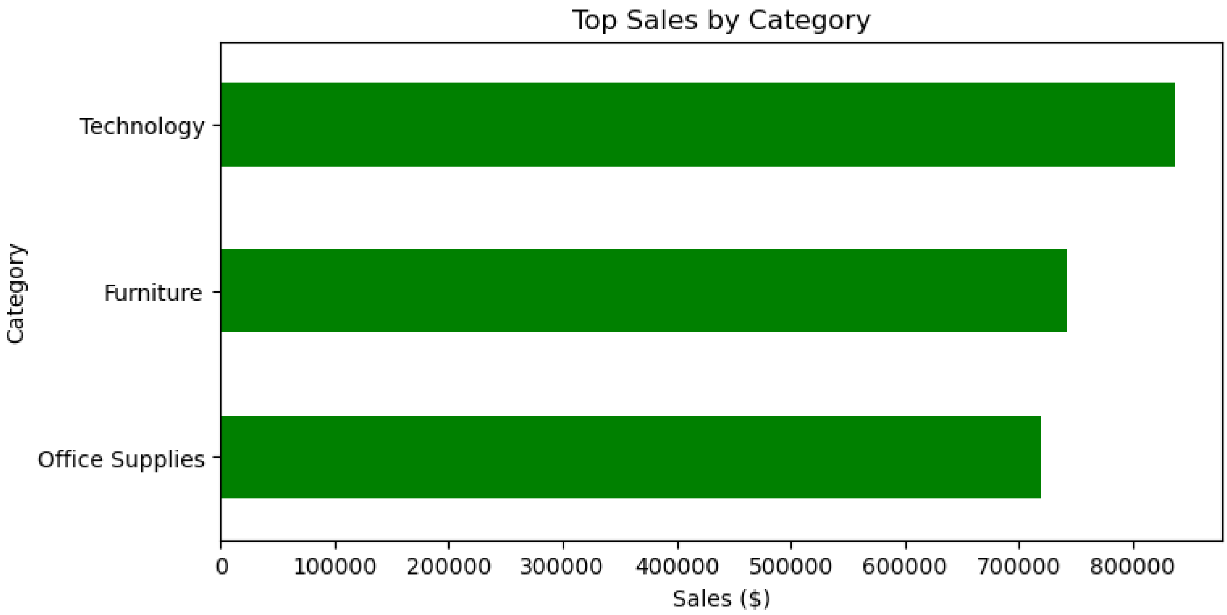
```
# Histogram of Sales
plt.figure(figsize=(8, 6))
sns.histplot(df["Sales"], bins=30, kde=True, color="blue")
plt.title("Sales Distribution")
plt.xlabel("Sales ($)")
plt.ylabel("Frequency")
plt.savefig("Sales Distribution")
plt.show()
```



```
#Top Categories by Sales
top_categories_sales = New_df.groupby('Category')
['Sales'].sum().sort_values(ascending=False)
top_categories_sales
```

```
Category
Technology        836154.0330
Furniture         741306.3133
Office Supplies   718735.2440
Name: Sales, dtype: float64
```

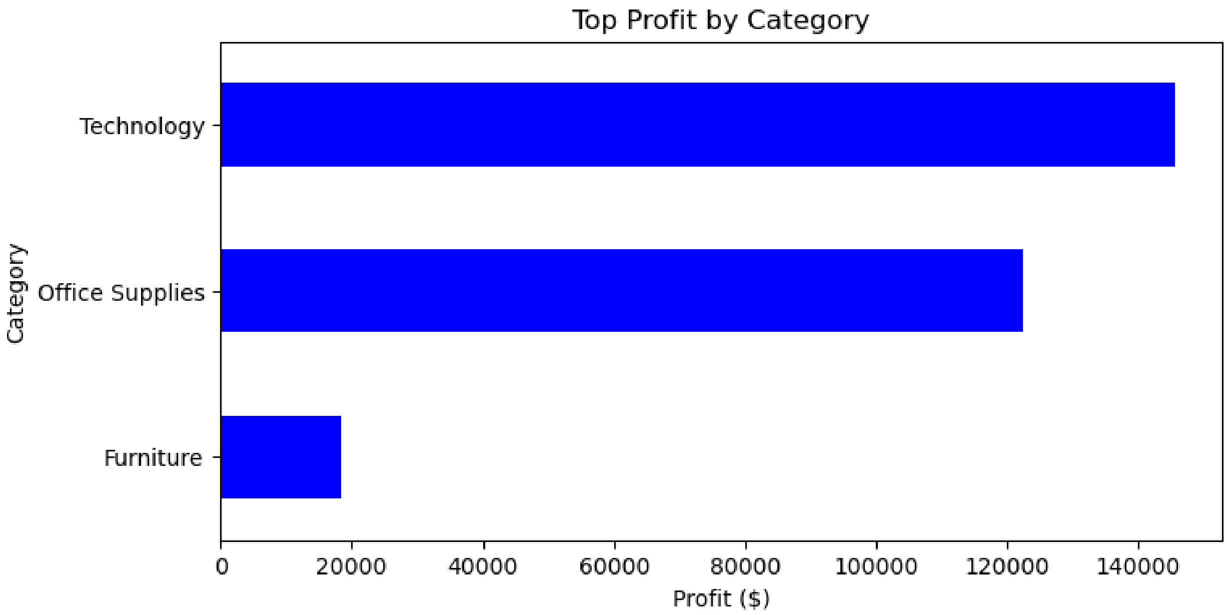
```
# Horizontal Bar Chart for Category Sales
plt.figure(figsize=(8, 4))
top_categories_sales.sort_values().plot(kind='barh', color='green')
plt.title("Top Sales by Category")
plt.xlabel("Sales ($)")
plt.ylabel("Category")
plt.savefig("Top Sales by Category.png") # Save the chart
plt.show()
```



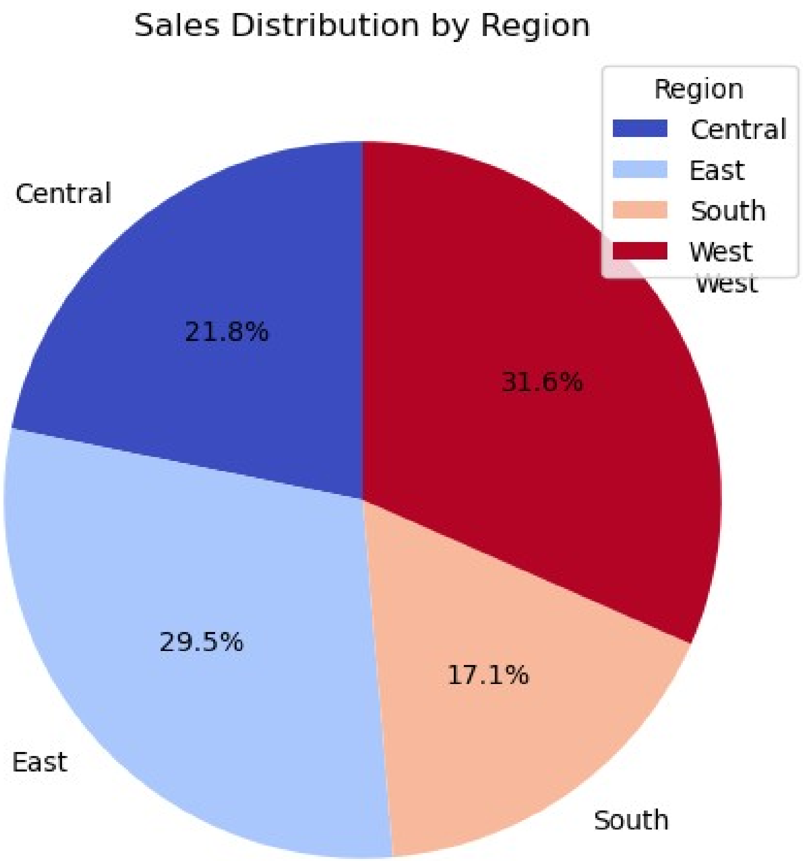
```
# Categorical Profit Analysis
profit_by_category = New_df.groupby('Category')
['Profit'].sum().sort_values(ascending=False)
profit_by_category
```

```
Category
Technology         145454.9481
Office Supplies    122364.6608
Furniture           18421.8137
Name: Profit, dtype: float64
```

```
# Horizontal Bar Chart for Category Profit
plt.figure(figsize=(8, 4))
profit_by_category.sort_values().plot(kind='barh', color='blue')
plt.title("Top Profit by Category")
plt.xlabel("Profit ($)")
plt.ylabel("Category")
plt.savefig("Top Profit by Category")
plt.show()
```



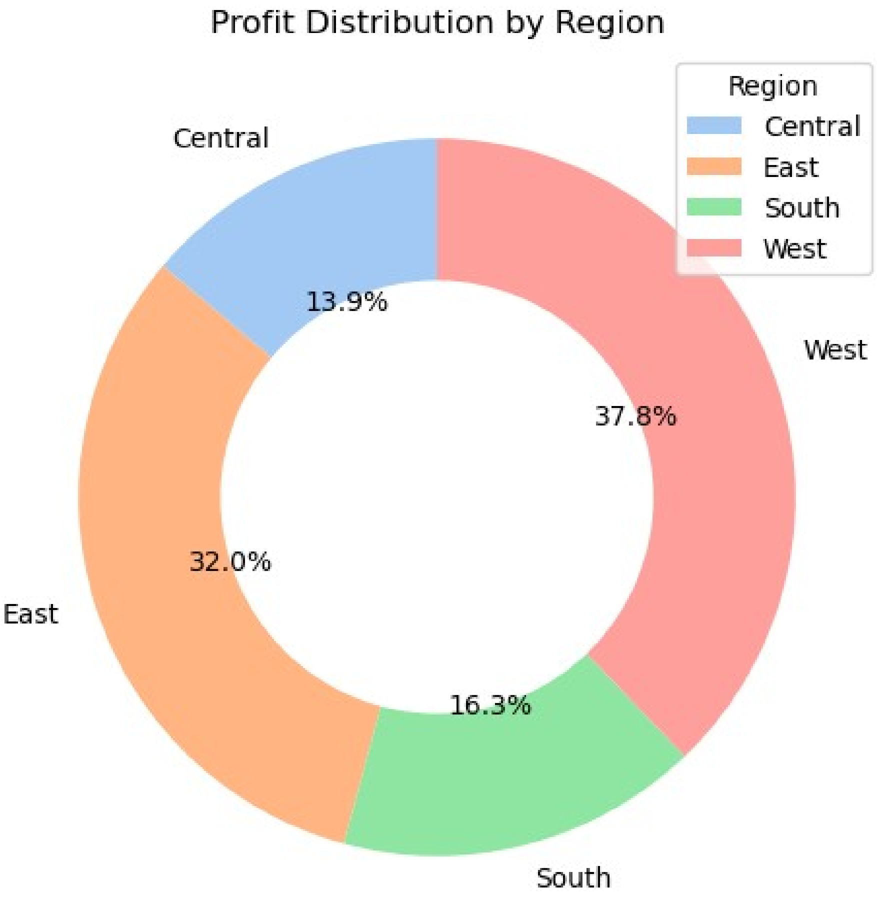
```
#Top Regional Sales
top_regional_sales = New_df.groupby('Region')
['Sales'].sum().sort_values(ascending=False)
top_regional_sales

Region
West       725255.6365
East       678435.1960
Central    500782.8528
South      391721.9050
Name: Sales, dtype: float64

# Pie Chart for Regional Sales
top_regional_sales = New_df.groupby('Region')['Sales'].sum()
plt.figure(figsize=(6, 6))
top_regional_sales.plot.pie(autopct='%1.1f%%',
startangle=90,labels=top_regional_sales.index, cmap="coolwarm")
plt.title("Sales Distribution by Region")
plt.legend(top_regional_sales.index, title="Region", loc="upper
right")
plt.ylabel("")  # Remove y-label
plt.savefig("Sales Distribution by Region")
plt.show()
```



```
#Top Regional profits
profits_by_region = New_df.groupby('Region')
['Profit'].sum().sort_values(ascending=False)
profits_by_region
```

```
Region
West       108329.8079
East        91506.3092
South       46749.4303
Central     39655.8752
Name: Profit, dtype: float64
```

```
# Pie Chart for Regional Profit
profits_by_region = New_df.groupby('Region')['Profit'].sum()
plt.figure(figsize=(6, 6))
profits_by_region.plot.pie(labels=profits_by_region.index,
autopct='%1.1f%%', startangle=90, colors=sns.color_palette("pastel"))
plt.gca().add_artist(plt.Circle((0, 0), 0.6, color='white'))  # Center
circle for doughnut effect
plt.title("Profit Distribution by Region")
plt.legend(profits_by_region.index, title="Region", loc="upper right")
plt.ylabel("")  # Remove y-label
plt.savefig("Profit Distribution by Region")
plt.show()
```

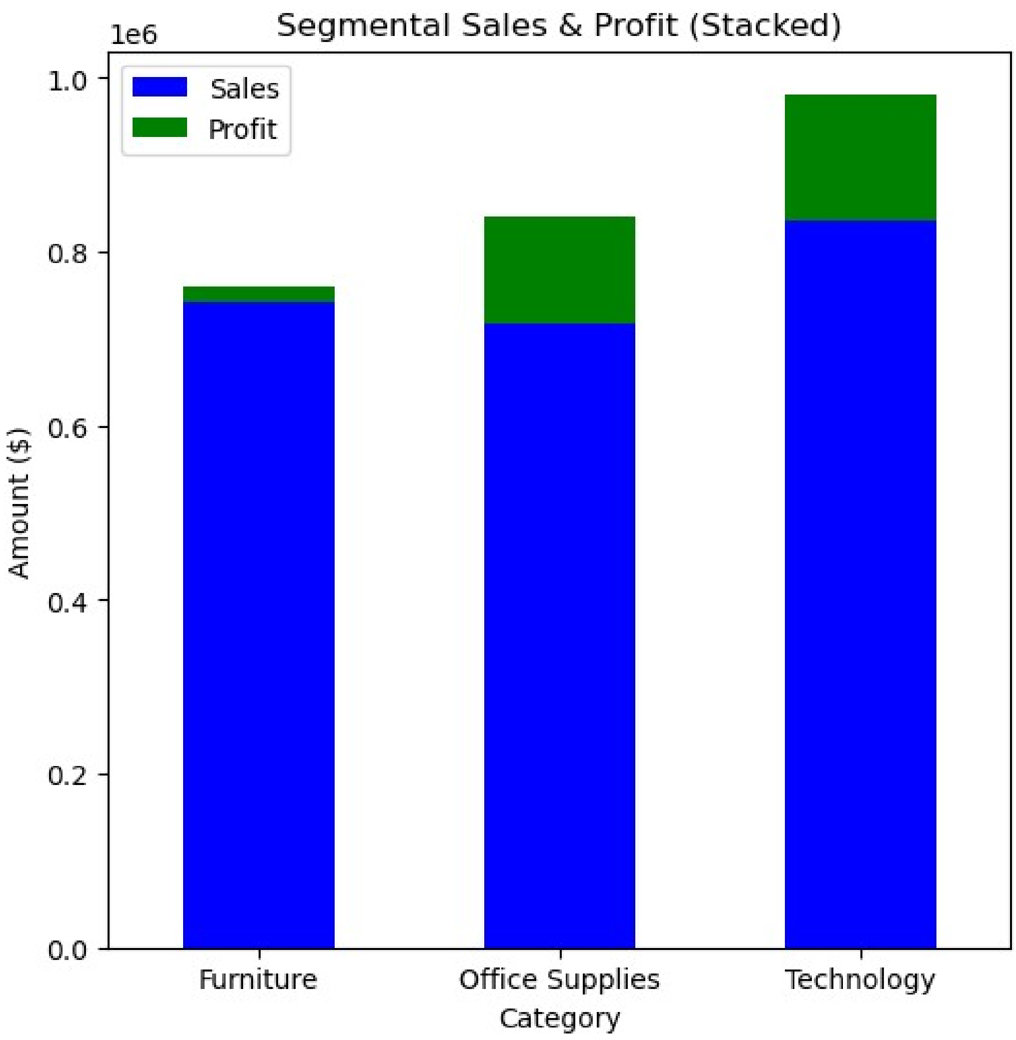


```
#Top Segmental Sales
top_segmental_sales = New_df.groupby('Segment')
['Sales'].sum().sort_values(ascending=False)
top_segmental_sales

Segment
Consumer       1.160833e+06
Corporate      7.060701e+05
Home Office    4.292927e+05
Name: Sales, dtype: float64

#Top Segmental Profit
profit_by_segment = New_df.groupby('Segment')
['Profit'].sum().sort_values(ascending=False)
profit_by_segment
```

```
Segment
Consumer       134007.4413
Corporate       91954.9798
Home Office     60279.0015
Name: Profit, dtype: float64
```

```
# Stacked Bar Chart for Category Sales & Profit
segment_sales_profit = New_df.groupby("Category")[["Sales",
"Profit"]].sum()
segment_sales_profit.plot(kind="bar", stacked=True, figsize=(6, 6),
color=["blue", "green"])
plt.title("Segmental Sales & Profit (Stacked)")
plt.ylabel("Amount ($)")
plt.xlabel("Category")
plt.legend(["Sales", "Profit"])
plt.savefig("Segmental Sales & Profit (Stacked)")
plt.xticks(rotation=0)
```

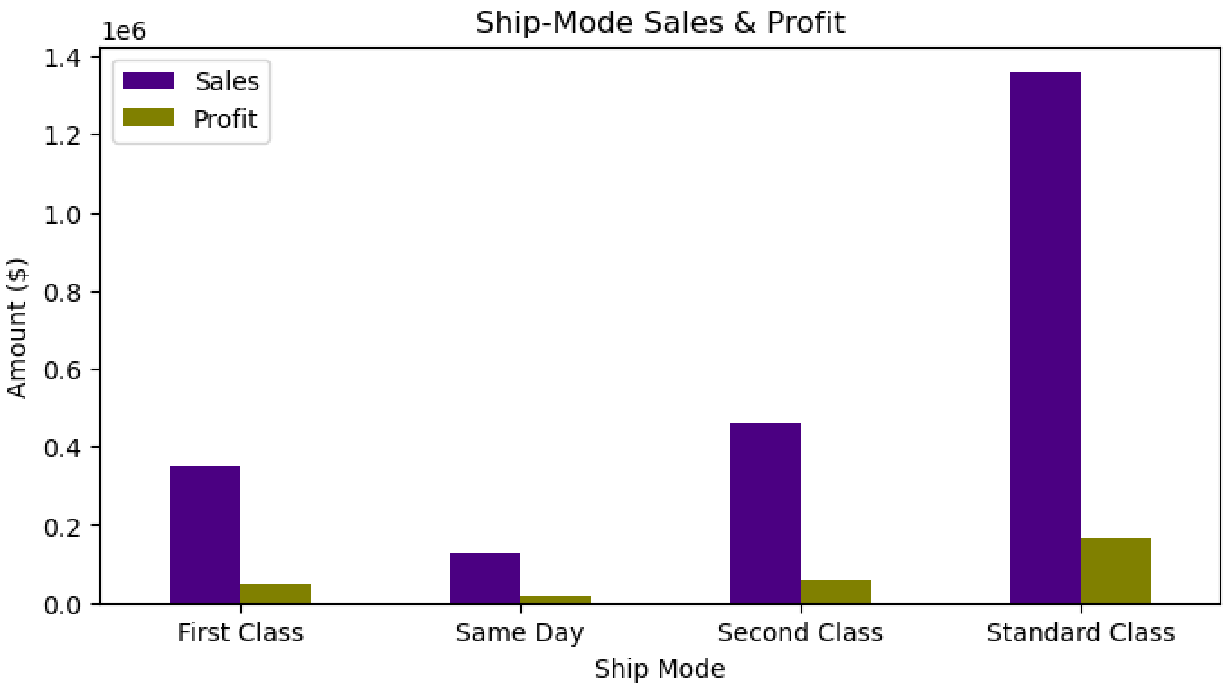
```
(array([0, 1, 2]),
 [Text(0, 0, 'Furniture'),
  Text(1, 0, 'Office Supplies'),
  Text(2, 0, 'Technology')])
```



```
#Top Ship Mode Sales
top_shipmode_sales = New_df.groupby('Ship Mode')
['Sales'].sum().sort_values(ascending=False)
top_shipmode_sales

Ship Mode
Standard Class    1.357316e+06
Second Class      4.591770e+05
First Class       3.513805e+05
Same Day          1.283217e+05
Name: Sales, dtype: float64

#Top Ship Mode Profit
profit_by_shipmode = New_df.groupby('Ship Mode')
['Profit'].sum().sort_values(ascending=False)
profit_by_shipmode
```

```
Ship Mode
Standard Class    163969.2280
Second Class       57446.6516
First Class        48953.6561
Same Day           15871.8869
Name: Profit, dtype: float64
```

```python
# Grouped Bar Chart for Ship Mode Sales & Profit
shipmode_sales_profit = New_df.groupby("Ship Mode")[["Sales",
"Profit"]].sum()
shipmode_sales_profit.plot(kind="bar", figsize=(8, 4),
color=["#4B0082" , "#808000"])
plt.title("Ship-Mode Sales & Profit")
plt.ylabel("Amount ($)")
plt.xlabel("Ship Mode")
plt.xticks(rotation=0)
plt.legend(["Sales", "Profit"])
plt.savefig("Ship-Mode Sales & Profit")
plt.show()
```

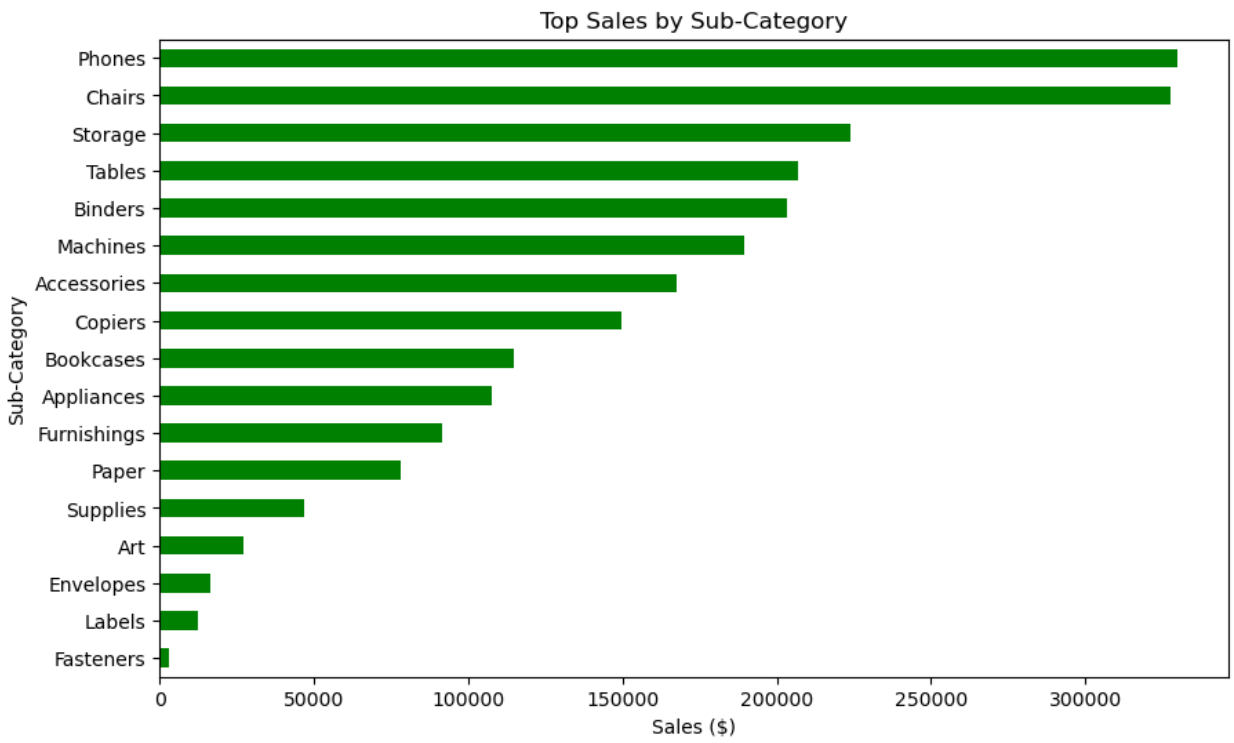


```python
# Import Data Exploratory Packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("SampleSuperstore.csv")
df
```

```
#Remove Duplicate
New_df = df.drop_duplicates()

#Top Sub-Category Sales
sub_category_sales = New_df.groupby('Sub-Category')
['Sales'].sum().sort_values(ascending=False)
sub_category_sales
```

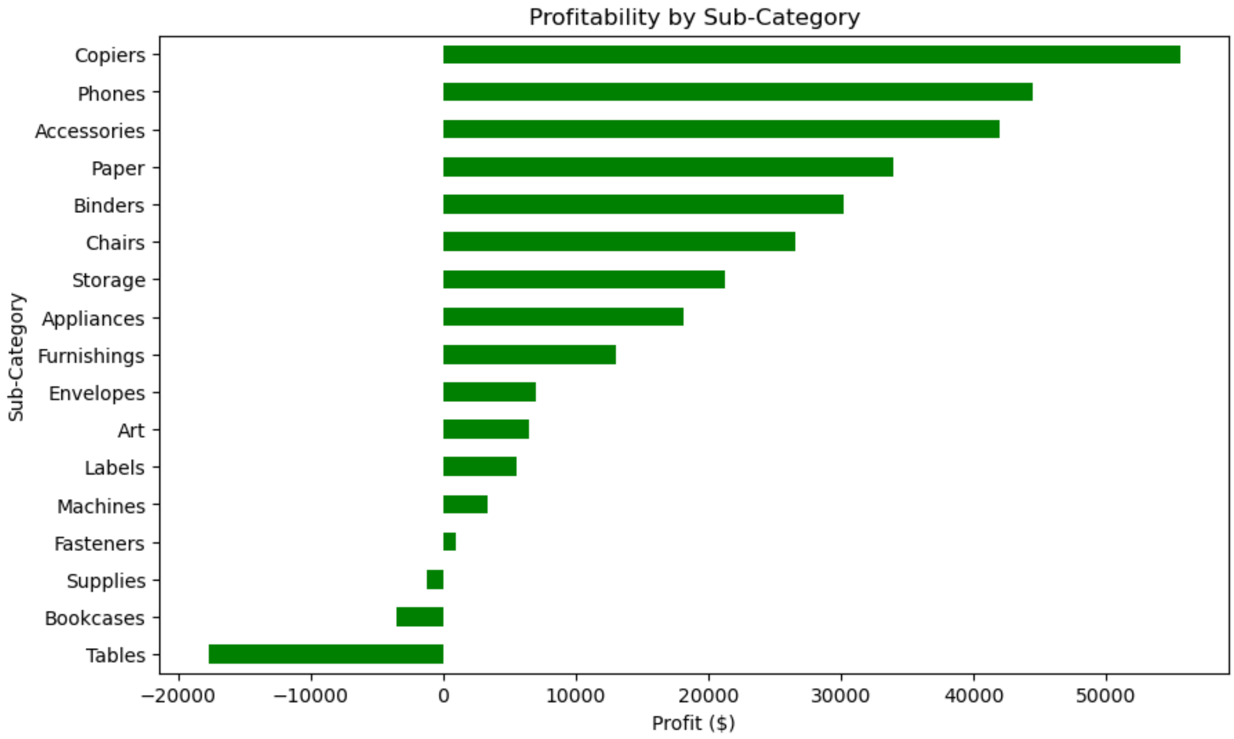
```
Sub-Category
Phones         330007.0540
Chairs         327777.7610
Storage        223843.6080
Tables         206965.5320
Binders        203409.1690
Machines       189238.6310
Accessories    167380.3180
Copiers        149528.0300
Bookcases      114879.9963
Appliances     107532.1610
Furnishings     91683.0240
Paper           78224.1420
Supplies        46673.5380
Art             27107.0320
Envelopes       16476.4020
Labels          12444.9120
Fasteners        3024.2800
Name: Sales, dtype: float64
```

```
# Horizontal Bar Chart for Sub-Category Sales
plt.figure(figsize=(10, 6))
sub_category_sales.sort_values().plot(kind='barh', color='green')
plt.title("Top Sales by Sub-Category")
plt.xlabel("Sales ($)")
plt.ylabel("Sub-Category")
plt.savefig("Top Sales by Sub-Category")
plt.show()
```

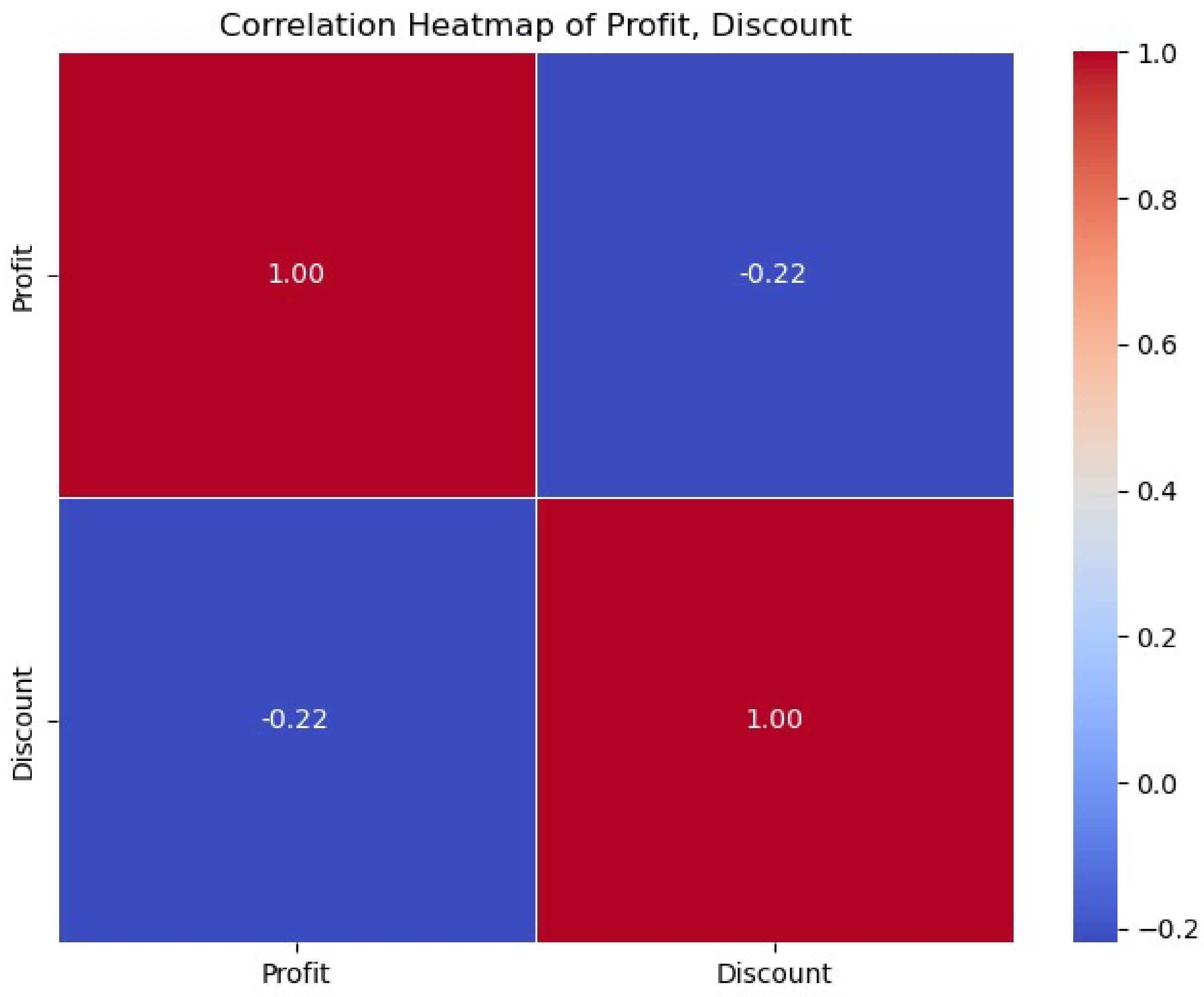


```
#Most and Least Profitable Sub-Category
sub_category_profits = New_df.groupby('Sub-Category')
['Profit'].sum().sort_values(ascending=False)
sub_category_profits

Sub-Category
Copiers        55617.8249
Phones         44515.7306
Accessories    41936.6357
Paper          33944.2395
Binders        30228.0003
Chairs         26567.1278
Storage        21278.8264
Appliances     18138.0054
Furnishings    13052.7230
Envelopes       6964.1767
Art             6524.6118
Labels          5526.3820
Machines        3384.7569
Fasteners        949.5182
Supplies       -1189.0995
Bookcases      -3472.5560
Tables        -17725.4811
Name: Profit, dtype: float64

# Horizontal Bar Chart for Sub-Category Profit
plt.figure(figsize=(10, 6))
```

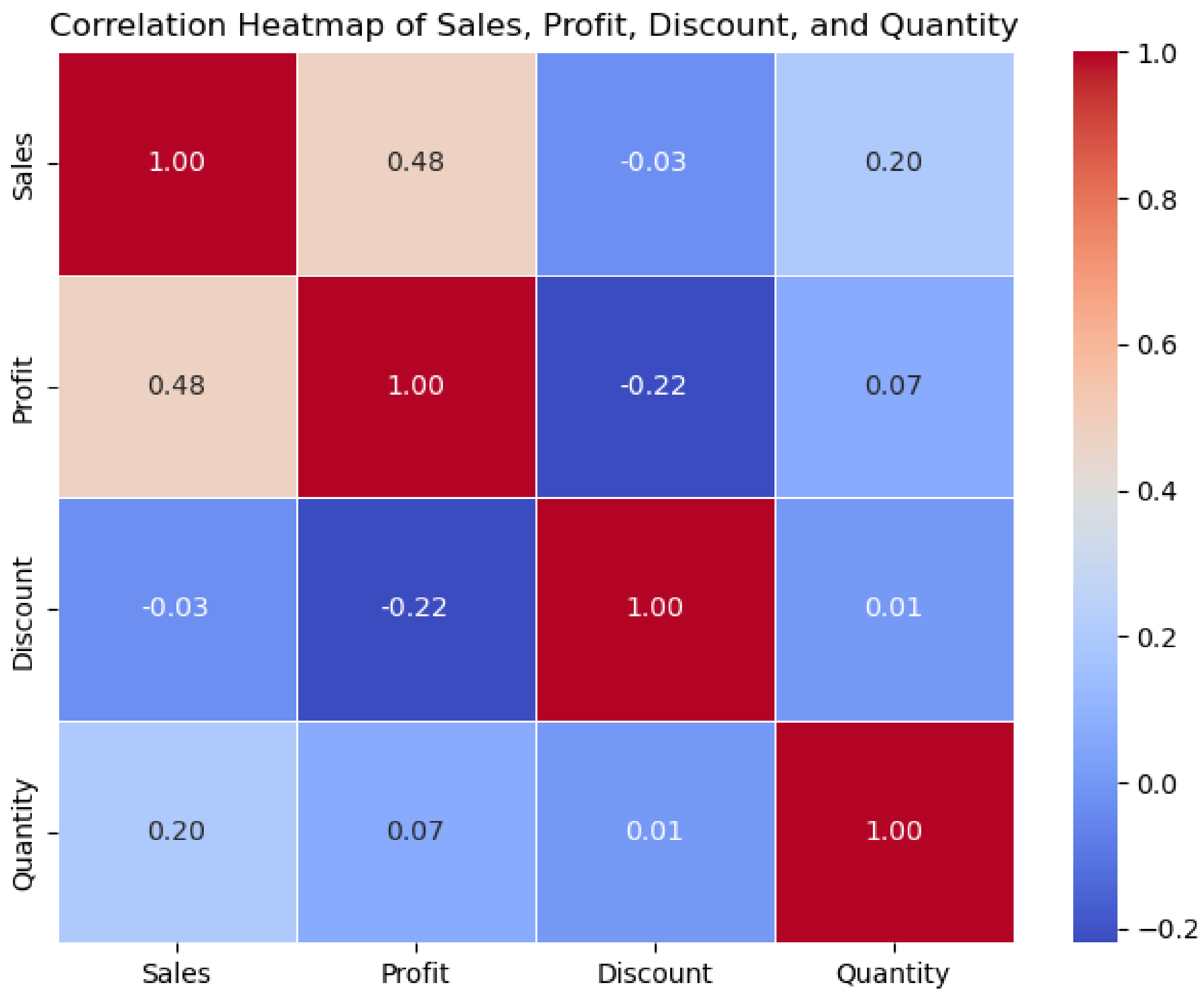
```
sub_category_profits.sort_values().plot(kind='barh', color='green')
plt.title("Profitability by Sub-Category")
plt.xlabel("Profit ($)")
plt.ylabel("Sub-Category")
plt.savefig("Profitability by Sub-Category")
plt.show()
```



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("SampleSuperstore.csv")
df
#Remove Duplicate
New_df = df.drop_duplicates()

# Compute correlation matrix
corr_matrix = New_df[["Profit", "Discount"]].corr()

# Heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f",
linewidths=0.5)
plt.title("Correlation Heatmap of Profit, Discount")
plt.show()
```



```
# Compute correlation matrix
corr_matrix = New_df[["Sales", "Profit", "Discount",
"Quantity"]].corr()

# Heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f",
linewidths=0.5)
plt.title("Correlation Heatmap of Sales, Profit, Discount, and
Quantity")
plt.savefig("Correlation Heatmap of Sales, Profit, Discount, and
Quantity")
plt.show()
```



```
#Correlation Between Discount and Quantity
corr_Discount_Quantity = New_df['Discount'].corr(New_df['Quantity'])
corr_Discount_Quantity
```

```
0.008678422048121645
```

```
#Correlation Between Sales and Profit
corr_Sales_Profit = New_df['Sales'].corr(New_df['Profit'])
corr_Sales_Profit
```

```
0.4790673141382176
```

```
# KPI Values
total_sales = New_df["Sales"].sum()
total_profit = New_df["Profit"].sum()

# Create a KPI Card with Matplotlib
fig, ax = plt.subplots(figsize=(8, 4))
ax.axis("off")
```

```
# Display KPI values
kpi_text = f"""
Total Sales: ${total_sales:,.2f}
Total Profit: ${total_profit:,.2f}
"""
ax.text(0.5, 0.5, kpi_text, fontsize=14, ha="center", va="center",
bbox=dict(boxstyle="round", facecolor="white", edgecolor="gray"))
```

```
Text(0.5, 0.5, '\nTotal Sales: $2,296,195.59\nTotal Profit:
$286,241.42\n')
```

Total Sales: $2,296,195.59
Total Profit: $286,241.42

```
import os

os.getcwd()
```

```
'C:\\Users\\pc'
```